Low Cost DCC (LC-DCC) 2.3

Controller Basics, Design and NMRA Compatibility

January 2026

Summary

Provided in this document are the basic system design details of an LC-DCC controller design that can be controlled via Windows PC, Linux PC, Android phone or tablet and/or using potentiometers for analogue control.

The hardware design is provided free but the software resident in the ARM processors or for the Windows and Android applications must be licensed by purchasing a license from eBay.

Disclaimer

The designers accept no responsibility for any damage to any train or accessory decoder connected to this DCC system through incorrect assembly or use of the hardware design.

Please read s-9.1_electrical_standards_2006.pdf NMRA standard before purchasing and using a power supply. Also note some cheap power supplies can give over voltage output.

Decoder Compatibility

Included at the end of this document is a list of decoders know to work with this LC-DCC controller. This list will increase with time as more and more people start using this LC-DCC controller for service mode programming and layout control.

Please let us know of any decoders not listed that are working with the LC-DCC control system.

License / Usage Terms

All the software components are protected by license. When you buy the software from eBay, you are entitled to one license which will be provided by Email, contact support@swws.co.uk with processor identification code. Extra licenses may be purchased via PayPal, go to http://www.swws.co.uk/lc-dcc.html and complete the PayPal order.

Existing users of the 1.X design

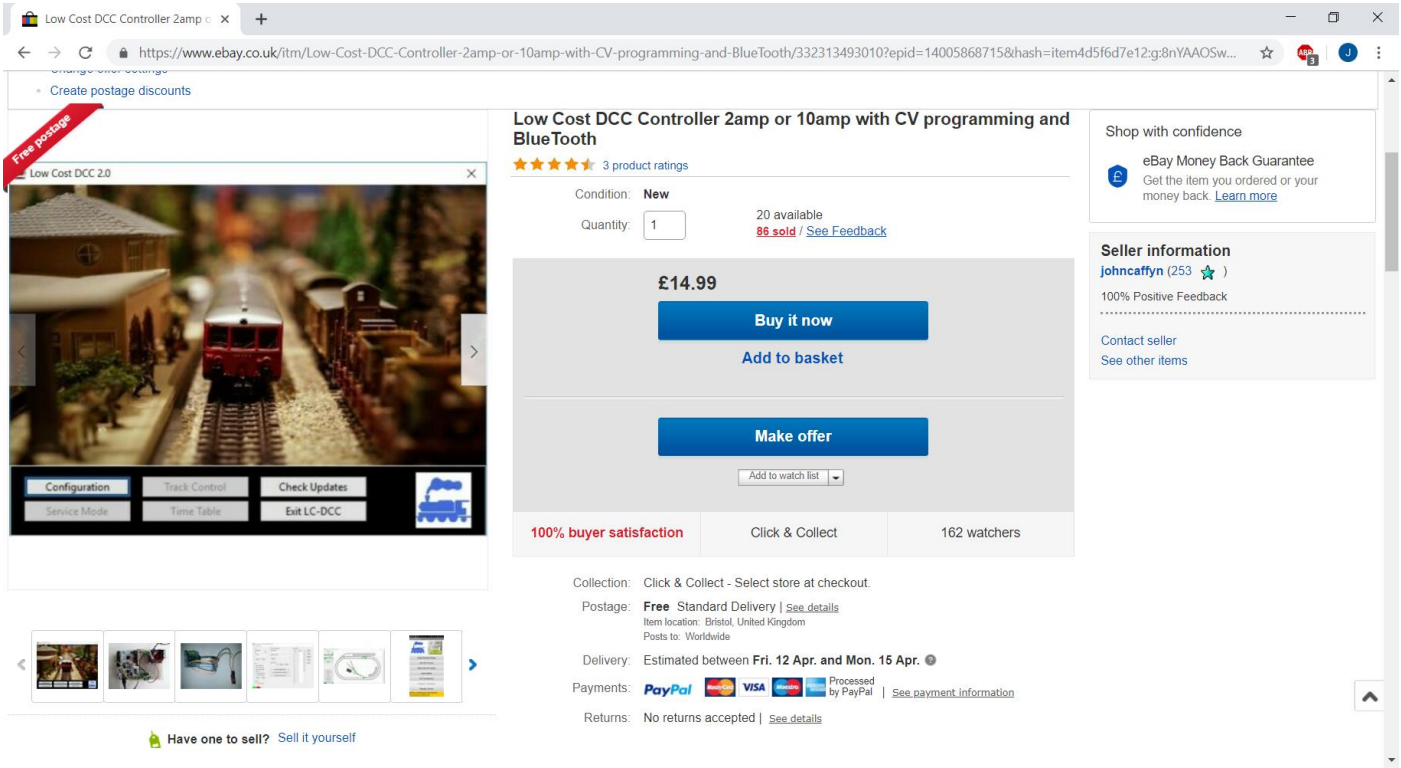
Anyone using version 1.X of this design should be aware that the function of M3/M4 pins A6/A7/PC0/PC1/PB0/PB1 has been swapped. Users should check the latest build documents for version 2.0 onwards. Existing users will need to obtain a license key, this will be provided free if you have purchased via the eBay account. Users creating their own software who want to upgrade will also need to contact support@swws.co.uk to obtain an “unlock code” for the M3/M4.

LC-DCC Forum

A low cost DCC forum can be found at http://low-cost-dcc.freeforums.net, this was created in March 2019.

Buying On EBay

Please note that there are a number of false adverts on eBay that cannot provide support or valid updates for this project and may not even provide anything for your money. The idea of Low Cost DCC is to make available a quality product at a price that everyone can afford. Only buy from the designer and developer of this project. To ensure you are buying from the only official eBay listing check the seller information is as displayed below (seller: johncaffyn, location: Bristol). Any other listing is false.



Contents

Introduction

This book describes how the LC-DCC controller system works, its NMRA compatibility and its basic concepts and operation. This book has been created to address feedback from current users.

The LC-DCC systems supports the following NMRA DCC features:

- 7-bit engine decoder addresses from 1 to 127
- 14-bit long engine decoder addresses from 1 to 10239 (0 is reserved)
- 9-bit accessory decoder address from 1 to 512
- 11-bit accessory decoder address from 1 to 2048
- 14, 28 or 128 engine speed steps selectable on an engine by engine basis
- Emergency stop
- F0 to F28 inclusive engine functions for light and sound
- Volume control for sound via extended DCC packet
- CV programming
- CV reading (requires an INA219 current monitor module)

The LC-DCC controller system supports the following features:

- Multiple users via USB serial, Bluetooth or WIFI
- Wireless train and point control via Bluetooth or WIFI
- Linux and Windows layout control applications via USB or WIFI
- Android Bluetooth application for train/accessory control and CV programming
- Voice controlled Android application via Bluetooth
- Up to 8 optional analogue speed controls via slider or rotary potentiometers and I2C or ADC
- Up to 64 digital outputs using PCF8574 I2C bus modules (Pico boards only)
- Current overload detection and automatic track power off with optional INA219 module
- Track A and track B outputs to allow booster operation or CV programming
- LC-DCC RFID[1] train location message forwarding is supported
- Software crash detection watchdog to protect train operation
- Stop switch for both engine stop and track power deactivation

The LC-DCC controller system does not support:

- Analogue train control, all engines must be fitted with DCC decoders

The LC-DCC controller is compatible with a number of NMRA standards as defined later in this document.

A list of DCC decoders that the system has been tested with is listed at the end of this document.

NOTE: 1. LC-DCC RFID system is under development and is an “add on” to the LC-DCC controller and will be available later as a separate item.

Licensing

The Windows, Linux and Android software require an activation code to work with the LC-DCC controller. The main reason for this is to allow all software to be downloaded from http://www.swws.co.uk/lc-dcc-files.html without the need to protect the software. As such every LC-DCC controller needs a different activation code. Multiple codes can be obtained with discount by contacting the EBay seller or sales@swws.co.uk.

When you first use either the Android software, the Linux software or the Windows software with a new LC-DCC controller you will need to obtain an activation code. The software will tell you that you need to contact support@swws.co.uk for a code and that you must provide them with the M3/M4/Pico processor identification displayed at the same time with the message.

You can build and test the basic M3/M4/Pico design without needing a code, but you will need an activation code to run trains, control accessories and program CVs.

Terminology

Here are some hardware terms used with the LC-DCC system and what they mean:

M3	An M3 micro controller (either blue pill board, black pill board or development board), the M3 communicates with the PC or Android device and creates DCC packets that are sent to the H-Bridge.
M4	An M4 micro controller (either black pill board or STM32F411 STM board), the M4 communicates with the PC or Android device and creates DCC packets that are sent to the H-Bridge.
Pico	A Raspberry Pi Pico and Pico2 micro controller (either pico or pico-w board from Raspberry Pi or third party suppliers), the Pico communicates with the PC or Android device and creates DCC packets that are sent to the H-Bridge.
INA219	An current monito module that measures current being used by the LC-DCC system. This module is required for current overload protection and reading CV values during programming. This module is optional, if not part of the LC-DCC system then there can be no overload detection or reading of CV decoder values (CV decoder values can still be written to however).
H-Bridge	This is a motor control module and provides the power to the tracks and generates the DCC packet signals that are sent to the DCC decoders in use. Depending of the required current to operate your layout, a different H-Bridge may be used in the design.
PCF8574	An I2C bus digital expander card. Up to 8 cards can be connected to the I2C bus, each with a different address. This provides 64 bits of digital output for controlling LEDs, relays etc.

The following terms relate to the design of the LC-DCC controller and it's use:

STATE/MODE	The LC-DCC system has three modes: IDLE, SERVICE and OPERATIONAL. In IDLE mode (default after power-up) the controller generates only DCC idle packets. In SERVICE mode the controller allows reading and writing to decoder CV values. In OPERATIONAL mode the controller allows locomotive speed and function control and also allows accessory decoder control.
LOCKS	The LC-DCC system allows both engine locks and accessory locks. These locks when active restrict engine control or accessory control to the first connection that uses them. This means when there are multiple users using the LC-DCC system each user cannot interfere with the other connected users. When there is only one user using multiple connections such as Bluetooth and USB these locks can be disabled.

CONNECTION The LC-DCC system allows USB, serial, Bluetooth and WIFI connections. Each connection is identified individually and the connection is used for STATE and LOCK control. When using WIFI, up to 5 connections are allowed to the controller.

STATE OWNER When the STATE/MODE is changed by any of the connections to the LC-DCC controller this connection becomes the STATE owner and only the state OWNER can change the STATE/MODE thereafter until the mode goes back to IDLE.

DCC Controller Designs

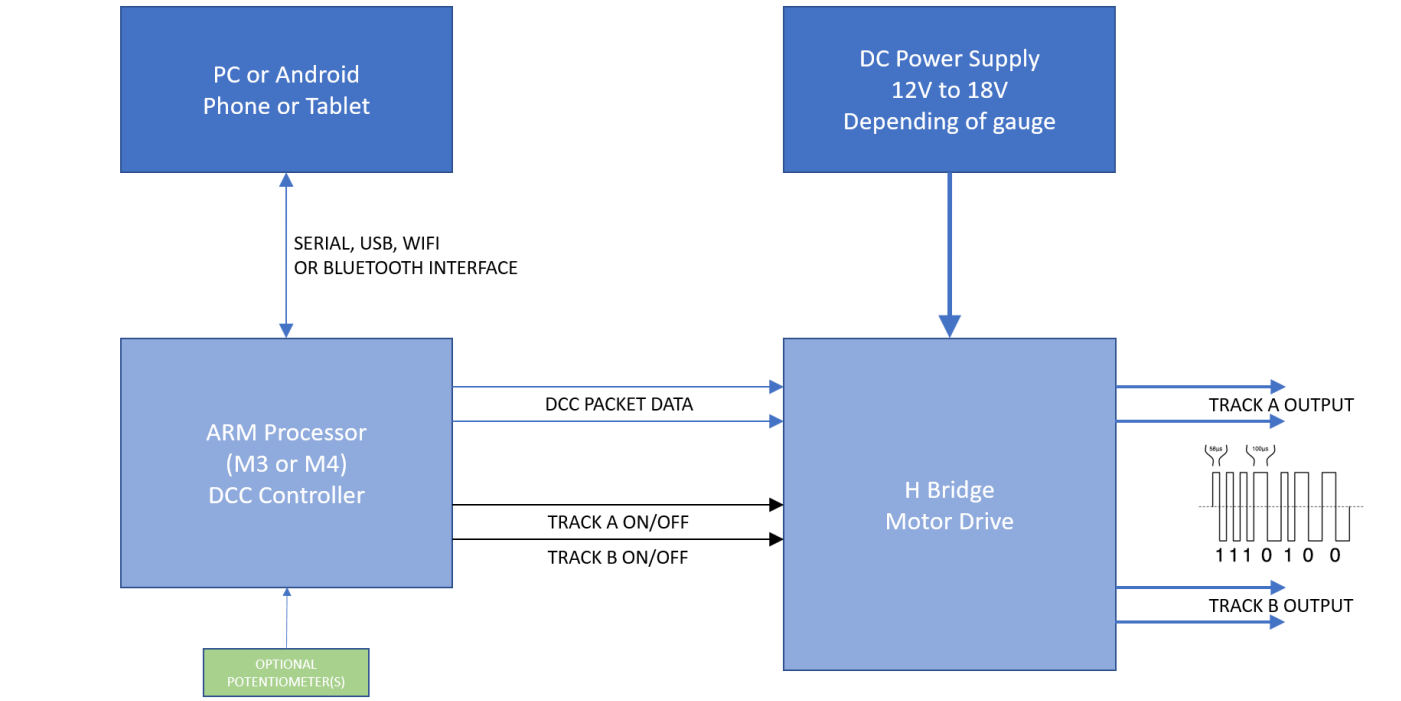
When building the LC-DCC system there are number different controller design options:

- M3 STM32F103 blue pill or black pill board-based controller
- M4 STM32F411 Nucleo board-based controller
- Raspberry Pi Pico or Pico-W controller supports Bluetooth, USB and potentiometers
- Controller with train control and CV programming (no INA219 current monitor)
- Controller with train control, CV programming, reading and overload detection (with INA219)
- Controller with eight analogue controls (using sliders or potentiometers)
- M3 STM32F103 with WIFI
- L298 2A+2A H bridge
- IBT-2 H bridge 43A (tested up to 10A with LC-DCC) suitable for all gauges
- IBT-4 H bridge 50A (tested up to 5A with LC-DCC) suitable for N/HO/OO and smaller gauges
- DRV8871 H bridge 3.6A

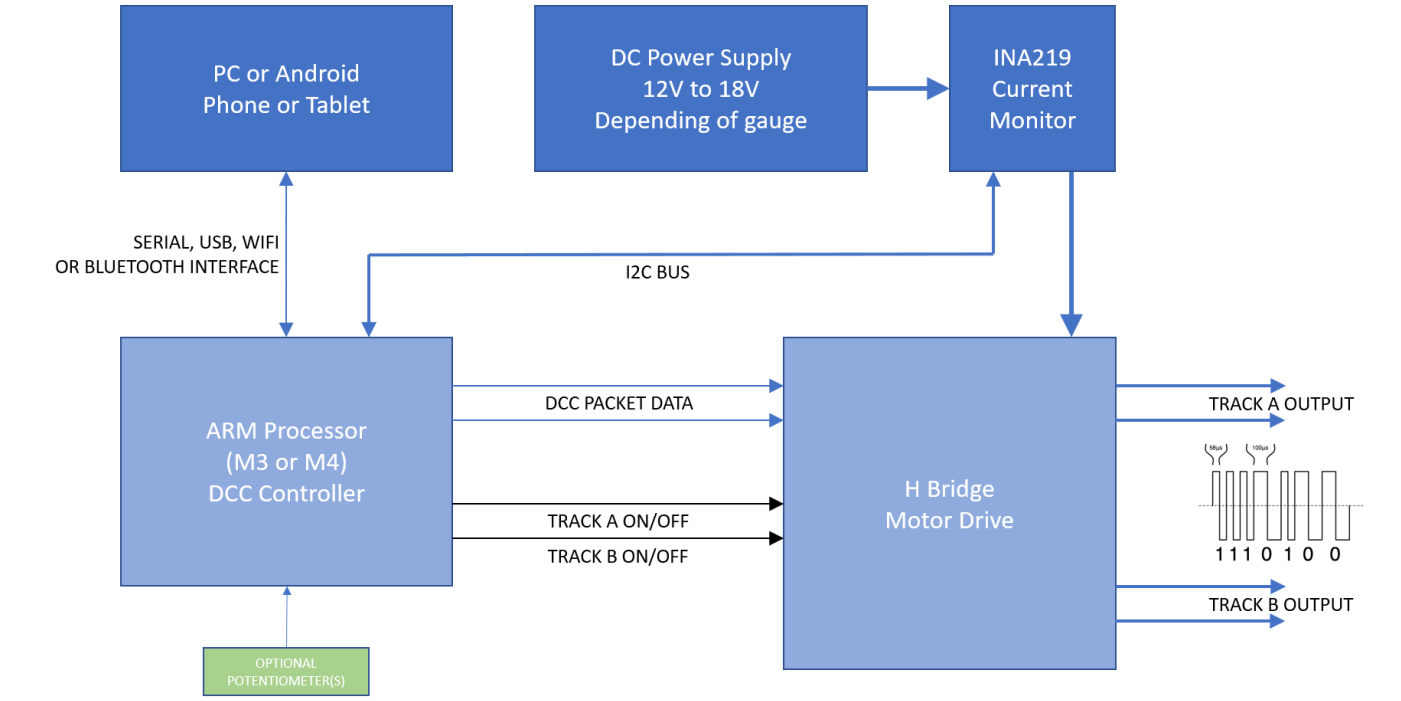
The main differences are:

- The M3 boards are cheaper than the M4 boards but need soldering skills
- The Raspberry Pi Pico is the cheapest solution but requires soldering skills
- The Raspberry Pi Pico-W is the cheapest Bluetooth solution but requires soldering skills
- Excluding the INA219 from the design removes overload detection and CV reading
- Adding the analogue controls will lose one UART used for Bluetooth or serial operation on M3
- WIFI allows five connections, serial and Bluetooth are limited to four connections maximum
- The L298 H bridge allows 2amp drive plus 2amp booster or CV programming track output
- The IBT-2 allows up to 10amp drive and can be combined with another IBT-2 or L298
- The DRV8871 H-bridge can support up to one 3amp drive output

The following block diagram shows a basic design with no INA219 for current monitoring:



This figure shows the basic design with an INA219 for current monitoring capability:



The software for both designs is the same, the DCC controller auto-detects the INA219 presence.

DCC Controller Operation

The M3, M4 or Pico ARM processor (DCC Controller) is used to generate the DCC packets that are sent to the tracks via the H-Bridge. The DCC Controller also controls whether the tracks are powered or not. Both of these functions are achieved by using digital output lines from the ARM processor micro controller connected to the H-Bridge.

If an INA219 is fitted the micro controller will detect this and use it to measure current when running trains or programming trains.

If the potentiometers are fitted via the ADS1115 I2C ADC modules then four (one ADS1115) or eight (two ADS1115) engines can be controlled using the potentiometers on the M3 or M4 processor. On the Raspberry Pi Pico there are three built-in ADC channels that can be used with potentiometers.

The DCC Controller has three modes or states of operation:

- Idle
- Operational
- Service Mode

In Idle mode the DCC Controller only generates DCC IDLE & RESET packets as per NMRA S 9.2 document. In service mode the DCC Controller only performs CV writing and optionally reading when an INA219 is fitted to the design. In operational mode the DCC Controller manages accessory decoders and engine decoders generating the DCC packets required to manage the trains and layout.

All LC-DCC controller mode changes are controlled via the USB/WIFI/Bluetooth/Serial interface of the micro controller. The same interface is used to read/write CV values, control trains and manage accessory decoders.

With WIFI and Bluetooth your train control interface is wireless and you can move around while using it.

Current designs can use either an STM32F103 M3 micro controller, an STM32F411RE M4 micro controller or a Raspberry Pi Pico RP2040 processor.

NMRA DCC Compliance

Engine Address

The LC-DCC system supports 7-bit multi-function decoder address range from 1 to 127.

The LC-DCC system supports 14-bit multi-function decoder address range from 1 to 10239, 0 is reserved.

Speed Steps

The LC-DCC system supports 28 speed steps (engine decoder CV 29 value bit 1 unset).

The LC-DCC system supports 128 speed steps (engine decoder CV 29 value bit 1 set).

Accessory Decoder Address

The LC-DCC system supports 9-bit accessory decoder address range from 1 to 512 and values 0 to 7. With the current software you can program on and off values (or left and right values) in the range 0 to 7.

Engine Decoder Functions

The LC-DCC system supports NMRA DCC function groups for FL (F0) and F1 to F28.

FL and F1 to F4 are implemented using NMRA DCC packet Function Group One Instruction (100) as described in NMRA standard S-9.2.1 July 2012.

F5 to F12 are implemented using NMRA DCC packet Function Group Two Instruction (101) as described in NMRA standard S-9.2.1 July 2012.

F13 to F20 are implemented using NMRA DCC packet Feature Expansion Instruction (110) as described in NMRA standard S-9.2.1 July 2012.

F21 to F28 are implemented using NMRA DCC packet Feature Expansion Instruction (110) as described in NMRA standard S-9.2.1 July 2012.

Extended Packet Volume Control

The LC-DCC system supports NMRA S-9.2.1 July 2012 “Analog Function Group” for volume control as described in the standard.

Power Supply Selection

The LC-DCC system does not come with a power supply. Please choose a DC power supply with a voltage suitable for your gauge. If in doubt consult the NMRA website for recommendations. The following table is a general guide:

Gauge	DC Supply Voltage
N	12.0V
OO/HO	14.4V
O/G	18.0V

Note however that modern engines appear to tolerate lower voltages than these. For example, some OO layouts can run on 12V it is dependent on voltage drop between the controller/booster and the engines.

Overload Detection

The LC-DCC system provides overload detection when an INA219 is fitted. This is achieved by monitoring the current using the INA219 and disabling the H Bridge drive if a current that is too high is detected.

The overload current is set via the Windows or Android applications.

Power Control Switch

Each controller design has a power control switch input. This accommodates a push button switch connected between 0V (NGD) and the switch input pin.

A short press of the push button switch will remove power for all track outputs. A long press (more than one second) will return track power outputs.

If an overload condition has been detected, this can be reset using the switch and a long press of the push button switch.

Multi User Operation

The LC-DCC system can support multiple users at any one time via either WIFI or multiple serial or Bluetooth connections.

The users cannot share engines and only one user can change the DCC controller state from idle to operational or service mode. Only one user can perform service mode operations.

The user that is first to select either operational or service mode is considered the state owner and only he or she can change the state of the DCC controller.

Once a user has selected an engine address and sent any command (speed/stop/function) he or she owns that address until the DCC controller returns to the idle state.

If a user attempts to control an already owned engine the Windows or Android application will indicate the engine address is locked.

Engine locking can be turned on or off using the Windows or Android application configuration window. The same is true for accessory locks. These options allow one user multiple connections to the DCC controller so walk about mode can be used.

Decoder Compatibility Table

Manufacturer	Model	Comments
LaisDCC	Engine decoder 2 function 2A/1A bare wire	No known issues[1]
LaisDCC	Engine decoder 4 function with NMRA 8 pin socket (860021)	No known issues[1]
LaisDCC	NEM 6 pin decoder	No known issues[1]
Hornby	R8249 Engine decoder	No known issues[1]
Hornby	R8247 Accessory decoder	No known issues[1]
Gaugemaster	6 pin engine decoders	No known issues[2]
Hornby	TTS Decoder	No known issues[2]
Bachmann	36-553	No known issues[1]
DCC Concepts	DCC CONCEPTS DCD-AD2FX COBALT IP ACCESSORY DECODER	No known issues[1]
Train$ave	TSV01a decoder	No known issues[2]
Digitrax	DCC Decoder	No known issues[2]
Atlas/Lenz	LE063XF N Gauge DCC Decoder	No known issues[2]
Digikeijs	DR4024 Servo Decoder	No known issues[1]
Train Tech	LFX6 Quad LED Lighting Controller	No known issues[1]
Dapol	Imperium Decoder (version 4)	No known issues[1]

1. These decoders have been tested by the DCC-CTRL developers.
2. These decoders are reported as working by users of the DCC-CTRL system.

The LC-DCC system has also been tested with the Arduino NMRA DCC library and has been used with the SMA20 Low Cost 17 Channel DCC Decoder Ver 6.01 found here: https://model-railroad-hobbyist.com/node/24316

Common Problems

CV Values are always read as zero in Service Mode

Check connections to programming track are good. Check power supply is on.

CV Values are read but appear incorrect

Check connections to programming track are good.

Engine moves during CV reading & writing

This is common as the engine powers the motor to signal acknowledge back to the DCC controller.

Hornby R8247 Decoder Not Working

When using the Hornby point decoders, one address is used for all four point controls. Each point control is switched using the values 0/1, 2/3, 4/5, 6/7. The outputs for point 1 use 0/1, point 2 use 2/3, point 3 use 4/5 and point 4 use 6/7. The 512 address AC command should be used to control the decoder.

These decoders can use four separate addresses for the four decoder point coil outputs. So when adding point controls to the track control layout you must add for each output two controls. One control direction must be set to 0 and the other set to 1 but they must have the same address. Also if the R8247 is programmed with a Hornby Select DCC controller, the address set on the controller will not be the DCC decoder address. For example if you program 61 on the Select controller this will program DCC decoder address 17 followed by 18, 19 and 20 for the four outputs. If the R8247 is used in one address mode then the output ports are switched by setting an accessory decoder value of 0/1 for output 1, 2/3 for output 2, 4/5 for output 3 and 6/7 for output 4. When programming decoder addresses using the DCC controller software remember to take one from the actual address you required before programming CV1 as the address range is 1..512 but the CV address range starts at zero (for address 1). M3/M4 Software Not Working Verify using the programming tools that the software has been loaded into flash. Look at the board LED and reset the board. The board LED should flash every second. INA219 Error If on any screen where the track current is displayed a message appears saying “INA219-ERROR” this indicates that the INA219 is not connect correctly to the processor board.

Programming Learning DCC Concepts Accessory Decoder Address

On the Windows/Linux track control screen there is a popup menu for sending accessory commands to an address to allow configuration of learning DCC Concepts accessory decoders. This should be done before any train control is attempted or the menu will become disabled.

TermTerm Or Windows Software Not Communicating With DCC Controller

Try re-installing the driver for the serial port being used on Windows.

Website References

Low Cost DCC Controller Software

Email: support@swws.co.uk

FAQS: http://www.swws.co.uk/lc-dcc-faqs.html

Downloads: http://www.swws.co.uk/lc-dcc-files.html

NMRA Website

https://www.nmra.org/

https://www.nmra.org/sites/default/files/s-92-2004-07.pdf

https://www.nmra.org/sites/default/files/s-9.2.1_2012_07.pdf

L298N

http://www.instructables.com/id/Arduino-Modules-L298N-Dual-H-Bridge-Motor-Controll/

INA219

https://www.adafruit.com/product/904

STM Nucleo F411RE

https://developer.mbed.org/platforms/ST-Nucleo-F411RE/

http://www.st.com/en/evaluation-tools/nucleo-f411re.html

http://www.st.com/en/embedded-software/stsw-link004.html

STM32F103

http://wiki.stm32duino.com/index.php?title=Blue_Pill

IBT-2

http://www.instructables.com/id/Motor-Driver-BTS7960-43A/

Pico & Pico-W

https:// www.raspberrypi.com/products/raspberry-pi-pico /

Version Change History

January 2026

Added PCF8574 digital output capability.

March 2025

Added power control switch detail. Added Raspberry Pi Pico board details.

September 2024

Added new 11-bit accessory command information and new Raspberry Pi Pico information.

April 2020

Added extra ADS1115 information to allow up to eight potentiometers for speed control. Added stop switch information. Added Android voice-controlled application information. Added Linux references.

April 2019

Initial version.